Amendments to the Specification:

Please replace the paragraph beginning at page 3, line 19, which starts with "For a particular HMM..." with the following amended paragraph:

For a particular HMM, the ~~Veterbi~~ Viterbi algorithm is used to find the most probable sequence of hidden states given a sequence of observed states. A ~~Veterbi~~ Viterbi block 26 receives inputs from sequence vector 22 and HMM table 24. HMM table 24 consists of three matrices for each word in the vocabulary, i.e. Initial State, State Transition, and Observation Probability Density Distribution. The Initial State matrix is a list of probabilities for starting in each of the possible states. The State Transition matrix lists the probabilities of transitioning from any given state to all possible states. The Observation Probability Density Distribution matrix lists the probabilities of any given speech unit being observed from any given state. ~~Veterbi~~ Viterbi block 26 provides an output that represents the single best state sequence or path to maximize the probability of having reached the desired state.

Please replace the paragraph beginning at page 4, line 38, which starts with "In operation, during the training process..." with the following amended paragraph:

In operation, during the training process for the speech recognizer 10, the speech unit sequence that makes up the current word is used to train the Hidden Markov Model (HMM) for that word. First the ~~Veterbi~~ Viterbi algorithm is used to determine the most probable state sequence for the model, given the speech unit sequence for the current word and given the word model to be trained. The ~~Veterbi~~ Viterbi algorithm performs a maximum likelihood computation by calculating a measure of similarity, or distance, for all possible paths or sequences. The state sequence is then used to update the state transition matrix and the observation probability density distribution matrix for the current HMM word model. The most probable state sequence is then recalculated based on the updated model, and then the model is updated again using the results of the recalculated state sequence. This update cycle is repeated until the model converges.

Once the model has converged for the speech input spoken by the current speaker, the speech inputs from the next speaker are processed and the HMM word model is updated again. The same HMM word model is repetitively updated for every speaker of the same vocabulary word.

A2

Please insert the following new paragraph before the paragraph beginning at page 6, line 8, which starts with "FIG. 4 is a flowchart...":

After the sample speech features are extracted, expanded and averaged, the resultant polynomial vectors are vector quantized during step 62 to provide a set of 4th order vectors that represent the desired number of speech building blocks for the target language and vocabulary of the recognizer. Vector quantization step 62 is described above with reference to vector quantizer 44 in Figure 2. The number of such vectors is defined by codebook size 64. After vector quantization step 62, a finite set of 4th order vectors represent the desired recognizer's vocabulary. The sum of the 4th order feature vectors is calculated during step 66. The individual feature vector of the current speech building block is scaled using the summation vector from step 66 and a scaling factor 70. The resulting scaled vector is then mapped into a square matrix during step 72. The square matrix is then decomposed (using a Cholesky decomposition) into an upper and a lower triangular matrix during step 74, and solved by back substitution during step 76. Steps 68 through 76 are repeated via next model step 78 until all speech building blocks have been processed and a model has been created for each. The resulting speech unit models are stored during step 80, and the process completes at step 82.

A3

Please replace the paragraph beginning at page 6, line 30, which starts with "During the training process for the speech recognizer..." with the following amended paragraph:

During the training process for the speech recognizer, the speech unit sequence that makes up the current word is retrieved from Word HMM 112 and used to train the Hidden Markov Model (HMM) for that word. The ~~Veterbi~~ Viterbi algorithm is used in process step 114 to determine the most probable state sequence for the model given the

A4

speech unit sequence for the current word and given the word model to be trained. The ~~Veterbi~~ Viterbi algorithm performs a maximum likelihood computation by calculating a measure of similarity, or distance, for all possible paths or sequences. Then, in process step 116 the state sequence is used to update the state transition matrix and the observation probability density distribution matrix for the current HMM word model. The most probable state sequence is then recalculated based on the updated model, and then the model is updated again using the results of the recalculated state sequence. This update cycle is repeated using process steps 114, 116 and 118 until the model converges. Once the model has converged for the current speaker the next speaker is processed and the HMM word model is updated again. The same HMM word model is repetitively updated for every speaker of the same vocabulary word as shown in process step 118.

A4

Please replace the originally filed Abstract with the following new abstract paragraph:

A5

A speech recognition system (10) having a sampler block (12) and a feature extractor block (14) for extracting time domain and spectral domain parameters from a spoken input speech into a feature vector. A polynomial expansion block (16) generates polynomial coefficients from the feature vector. A correlator block (20), a sequence vector block (22), an HMM table (24) and a ~~Veterbi~~ Viterbi block (26) perform the actual speech recognition based on the speech units stored in a speech unit table (18) and the HMM word models stored in the HMM table (24). The HMM word model that produces the highest probability is determined to be the word that was spoken.